Chapter 11
Implementation and Interaction of Estimators and Classifiers

Jon Hamkins and Hooman Shirani-Mehr

In each of the previous chapters, a method was proposed to estimate or classify a given signal parameter based on observations of the received signal. When tractable, the estimation or classification was derived from the maximum-likelihood (ML) principle, i.e., the parameter was estimated in the way that would best explain the observations. When ML solutions were impractical, reduced-complexity approximations to the ML solutions, or ad hoc estimators/classifiers were proposed.

In this final chapter, we explain how the algorithms of the previous chapters may be incorporated into a single, practical, and operational autonomous radio. This chapter is the bookend to Chapter 1, once again addressing the overall architecture of the autonomous radio and the interactions of its components. In particular, we summarize the algorithms that result from the analysis of the earlier chapters, show their interdependence, and construct an explicit sequence of coarse and fine estimation/classification that accomplishes all the functions of an autonomous radio. We have used the technique outlined in this chapter to write a software implementation of an autonomous radio that successfully identifies and processes an Electra-like signal with unknown attributes.

The chapter is organized as follows. In Section 11.1, we review the conventional approach to converting continuous-time signal processing to the discrete-time processing appropriate for a software implementation. In Section 11.2, we review a sequence of estimator/classifier actions that provides the first, coarse estimates of signal parameters, and a technique to refine the estimates by feeding back the coarse estimates to all the estimator modules. The individual estima-

tion/classification modules are discussed in the order in which they process the signal, including modulation index estimation in Section 11.3.1.

11.1 Signal Model

The complex baseband representation of the received signal is given by Eq. (1-8), which we restate here:

$$\tilde{r}(t) = \sqrt{2P_d} \sum_{l=-\infty}^{\infty} d_l(t) p[t - lT - \varepsilon T] e^{j[\omega_r t + \theta_c]} + \sqrt{2P_c} e^{j[\omega_r t + \theta_c(t)]} + \tilde{n}(t) \quad (11\text{-}1)$$

In order to process this continuous-time signal digitally, we sample the signal at regular time intervals, separated by T_s seconds.

Although there is nothing new about sampling a continuous-time signal, it is helpful to be explicit about the variance of the noise samples, which is related to the bandwidth of Eq. (1-1). If we follow the convention we have used throughout the monograph that $\tilde{n}(t) = \sqrt{2}\big(n_c(t) + j n_s(t)\big)$ and that each of $n_c(t)$ and $n_s(t)$ has two-sided power spectral density (PSD) $N_0/2$, we quickly see that a *sample* $\tilde{n}(kT_s)$ will have variance $R_{\tilde{n}}(0) = 2N_0\delta(0)$ per dimension, which is not bounded.

Instead, following standard practice [1], we implicitly assume that the passband signal $r(t)$ in Eq. (1-1) can be sent through an ideal passband filter that introduces negligible distortion to the signal but which eliminates the noise frequency components outside the passband. If the filter has bandwidth ω_s and is centered at the carrier ω_c, then the passband noise $n(t)$ at the filter output has PSD

$$S_n(\omega) = \begin{cases} N_0/2, & |\omega - \omega_c| \le \omega_s/2 \text{ or } |\omega + \omega_c| \le \omega_s/2 \\ 0, & \text{otherwise} \end{cases} \quad (11\text{-}2)$$

In complex baseband, the PSD of each of $n_c(t)$ and $n_s(t)$ is given by

$$S_{n_c}(\omega) = S_{n_s}(\omega) = \begin{cases} N_0/2, & |\omega| \le \omega_s/2 \\ 0, & |\omega| > \omega_s/2 \end{cases} \quad (11\text{-}3)$$

(Note that the PSD of each of $\sqrt{2}n_c(t)$ and $\sqrt{2}n_s(t)$ is twice this amount.) If $\omega_s = 2\pi/T_s$, then based on the above, each passband noise sample $n(kT_s)$ has variance N_0/T_s, and each complex baseband noise sample $\tilde{n}(kT_s)$ also has variance N_0/T_s in each dimension.

Thus, we model the discrete-time complex baseband signal as

$$\tilde{r}[k] \triangleq \sqrt{2P_d} \sum_{l=-\infty}^{\infty} d_k(kT_s)p[kT_s - lT - \varepsilon T]e^{j[\omega_r kT_s + \theta_c]}$$

$$+ \sqrt{2P_c}e^{j[\omega_r kT_s + j\theta_c(kT_s)]} + \tilde{n}(kT_s) \tag{11-4}$$

where each noise complex sample $\tilde{n}(kT_s)$ has variance N_0/T_s per dimension.

After reception of the baseband signal in Eq. (11-4), the signal is separated into its data-modulated and residual carrier components. We assume that, when a residual carrier signal is used, its spectrum is distinct enough from that of the data-modulated portion of the signal that these two components may be ideally separated. A high-pass filter extracts the data-modulation component, while a low-pass filter (LPF) extracts the residual carrier component:

$$\tilde{r}_d[k] \triangleq \sqrt{2P_d} \sum_{l=-\infty}^{\infty} d_k(kT_s)p[kT_s - lT - \varepsilon T]e^{j[\omega_r kT_s + \theta_c]} + \tilde{n}'(kT_s) \tag{11-5}$$

$$\tilde{r}_c[k] \triangleq \sqrt{2P_c}e^{j[\omega_r kT_s + j\theta_c(kT_s)]} + \tilde{n}''(kT_s) \tag{11-6}$$

The complex process $\tilde{n}'(kT_s)$ is the same as $\tilde{n}(kT_s)$ except that it has had a notch of spectrum removed. Generally the notch is small relative to the overall bandwidth, and so the complex sample $\tilde{n}'(kT_s)$ can be assumed to have variance N_0/T_s per dimension, as before.

The situation for $\tilde{n}''(kT_s)$ is a little more complicated. If the low-pass bandwidth ω_{lp} satisfies $\omega_{lp} < 2\pi/T_s$, then sampling at rate T_s results in a correlated $\tilde{n}''(kT_s)$ noise sequence. To avoid this situation, we may sample the residual carrier component at a different frequency, every $T_c = \omega_{lp}/(2\pi)$ seconds. Redefining $r_c[k]$ in this way results in

$$\tilde{r}_c[k] \triangleq \sqrt{2P_c}e^{j[\omega_r kT_c + j\theta_c(kT_c)]} + \tilde{n}''(kT_c) \tag{11-7}$$

where now $\{\tilde{n}''(kT_c)\}$ is an *uncorrelated* complex noise sequence with variance N_0/T_c per dimension.

11.2 Interaction of Estimator and Classifiers

Chapter 1 described a sequence of operations that accomplishes the task of estimating all the signal parameters shown in Table 11-1. This order is summarized in Fig. 1-3. We now become more explicit about the inputs and outputs from each of the modules and how they are connected.

Table 11-1. Signal parameters to be estimated and classified.

Notation	Description
β	Modulation index
ω_r	Residual frequency
T	Symbol duration
$p(t)$	Data format/pulse shape
R	Signal-to-noise ratio
ε	Fractional symbol timing
θ_c	Carrier phase
M	Modulation order (in M-PSK signals)

The autonomous radio begins in a *coarse estimation* phase in which the signal flow is unidirectional, without feedback. In coarse estimation, shown in Fig. 11-1, each estimation module has parameter estimates from the modules to its left, but no parameter estimates from the modules on its right. For example, the modulation index estimator operates without knowledge of the modulation type. Of course, every module has access to the $\tilde{r}[k]$ observables. Thus, in coarse estimation we employ the algorithms that require the least parameter information, which results in worse performance as compared to conventional receivers that rely on known attributes of the signal.

After the coarse estimation phase, estimates of all signal parameters are available to all blocks in the subsequent iterations. Therefore, the better-performing estimation techniques can be used in the *fine estimation* phase. For example, the modulation index estimator can make use of the modulation order $\hat{M}$, the data-transition tracking loop (DTTL) can make use of $\hat{\theta}_c$ to operate coherently, and so on. In the fine estimation phase, the modules can iteratively update their estimates until convergence takes place.

A complete functional diagram of both the coarse and fine estimation phases is shown in Fig. 11-2. In coarse estimation, the switches are in the "coarse" position. After all parameter estimates are available, the switches may be placed in the "fine" position. The received discrete-time signal is shown on the left. It enters the modulation index classifier and then, depending on the result, the appropriate structure is used to correct any residual frequency. Following this, the residual carrier (if any is detected) and data components are separated using low- and high-pass filters, and the joint estimator for data rate, data format, signal-to-noise ratio (SNR), and coarse symbol synchronization (sync) is run. These parameters, along with the carrier and data signal, are fed to the carrier

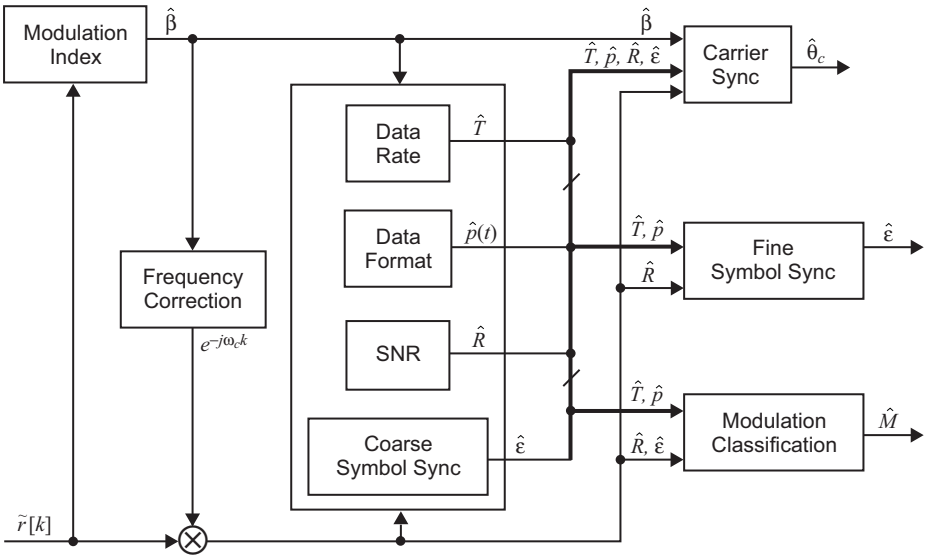


Fig. 11-1. Signal flow in coarse estimation.

sync, fine symbol sync, and modulation classifier structures. Depending on the modulation index, one of several carrier loops may be chosen. When carrier lock is detected, the coherent versions of any of the estimators may be used, as indicated in Fig. 11-2.

11.3 Coarse and Fine Estimators/Classifiers

In this section, we summarize the operation of the individual modules as they operate in coarse and fine modes.

11.3.1 Modulation Index Estimation

In the coarse estimation phase, the modulation index estimator requires no parameter estimates, other than the minimum symbol period T^*. It directly estimates the carrier and data powers by integrating over a sufficient epoch. In the fine estimation phase, it may operate coherently and use knowledge of the date rate, modulation type, symbol boundaries, and pulse shape to improve its performance.

The coarse estimator is given by Eq. (3-49), which becomes, after transforming to discrete-time,

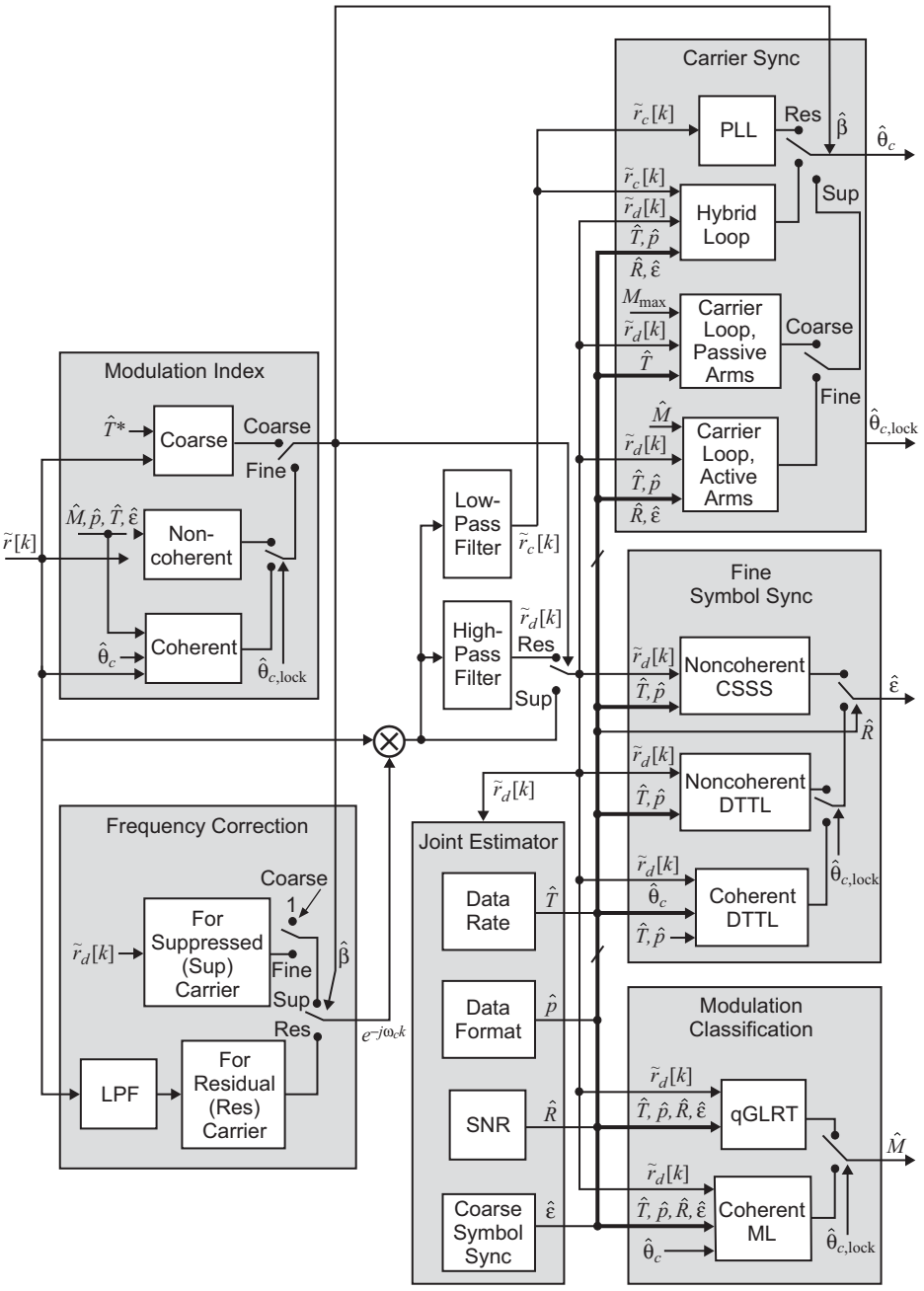


Fig. 11-2. A more detailed functional diagram of the autonomous radio architecture.

$$\hat{\beta} = \cot^{-1}\left[\sqrt{\frac{2\left[\left(\sum_{k=0}^{K'-1} y_c[k]\right)^2 + \left(\sum_{k=0}^{K'-1} y_s[k]\right)^2\right]}{K'\sum_{k=0}^{K'-1}\left[\left(y_c[k] - y_c[k - T^*/T_s]\right)^2 + \left(y_s[k] - y_s[k - T^*/T_s]\right)^2\right]}}\right] \tag{11-8}$$

where $K' = KT^*/T_s$, and $y_c[k] = y_c(kT_s)$ and $y_s[k] = y_s(kT_s)$, which are defined in Eq. (3-42).

In the fine estimation phase, we may replace T^* in Eq. (11-8) by the estimate $\hat{T}$ at the output of the data rate estimator. Also, timing estimates will allow us to define y_c and y_s as in Eq. (3-32) instead of as in Eq. (3-42), i.e., with the symbol-timing offset removed, which improves the fidelity of summations in Eq. (11-8).

If the modulation type is estimated to be binary phase-shift keying (BPSK) in the coarse phase, we may improve the modulation index estimate in the fine estimation phase by using the modulation index estimator for coherent BPSK in Eq. (3-9), which in discrete-time becomes

$$\cot\hat{\beta} = \frac{\sum_{k=0}^{K\hat{T}/T_s-1} y_s[k]}{\sum_{k=0}^{K-1}\left|\sum_{l=k\hat{T}/T_s}^{(k+1)\hat{T}/T_s} y_c[l]p\left[l - k\hat{T}/T_s\right]\right|} \tag{11-9}$$

If the carrier tracking loop is not yet in lock, then the noncoherent BPSK modulation index estimator from Eq. (3-39) may be used. If the modulation type is a higher-order M-ary phase-shift keying (M-PSK), we may use a discrete-time version of Eq. (3-29).

11.3.2 Frequency Correction

Frequency correction is performed in the coarse estimation phase only if the modulation index estimator determines that the signal contains a residual carrier ($\hat{\theta}_c < \pi/2$). In that case, the technique of Section 4.1 can be applied. The scheme for a residual carrier signal is illustrated in Fig. 11-3.

In the fine estimation phase, when the symbol rate $1/T$, fractional symbol timing ε, and pulse shape $p(t)$ are known, the techniques of Section 4.2 or Section 4.3 may be used. Figure 11-4 illustrates one of these schemes. The input to the block diagram is the output of a matched filter, which requires prior knowledge of T, M, ε, and $p(t)$.

Examples of the acquisition and tracking performance are illustrated in Fig. 11-5. In Fig. 11-5(a), there is no residual frequency, and this is tracked

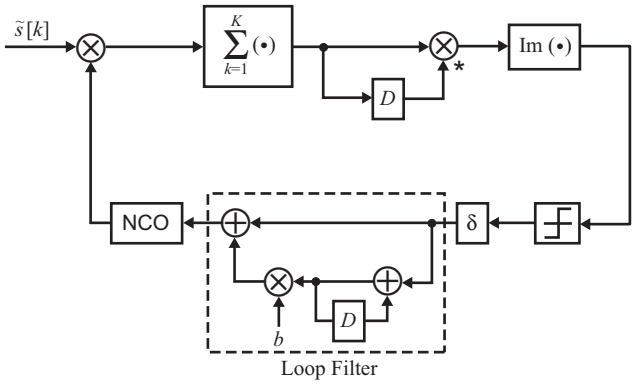


Fig. 11-3. Closed-loop frequency correction for a residual carrier signal (NCO = numerically controlled oscillator).

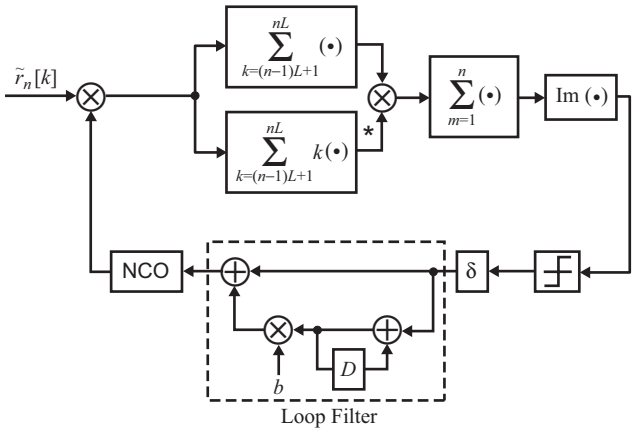


Fig. 11-4. Closed-loop frequency correction for a suppressed-carrier signal.

quite readily. In Fig. 11-5(b), the residual frequency is a half percent of the sample rate, and we see a short acquisition period followed by active tracking. In Fig. 11-5(c), the residual frequency is one percent of the sampling rate, and we see a longer acquisition period.

11.3.3 Joint Estimation of Data Rate, Data Format, SNR, and Coarse Symbol Timing

As shown in Fig. 11-2, the joint estimator for data rate, data format, SNR, and coarse symbol synchronization operates in the same way during both coarse and fine estimation phases. This is a consequence of the fact the the split-symbol moments estimator (SSME) for SNR estimation is independent of both

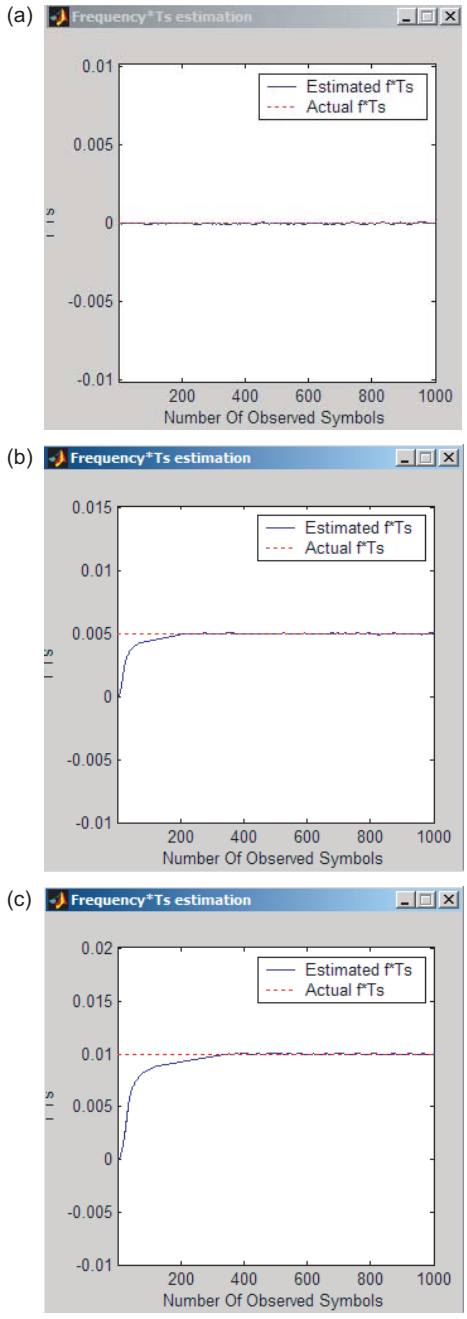


Fig. 11-5. Dynamic response of the frequency-tracking loop for a signal with SNR = 10 dB, $\theta_c = \pi/4$**: (a)** $\omega_r T_s/(2\pi) = 0$**, (b)** $\omega_r T_s/(2\pi) = 0.005$**, and (c)** $\omega_r T_s/(2\pi) = 0.01$**.**

the M-PSK modulation order and the carrier phase θ_c. This module takes the frequency-corrected version of $\tilde{r}[k]$ as input and produces $\hat{T}$, $\hat{p}(t)$, $\hat{R}$, and $\hat{\varepsilon}$ as output.

Figure 11-6 illustrates an example of the performance of the SNR estimator, assuming the residual frequency and symbol timing are known ($\omega_r = 0$ and $\varepsilon = 0$). The simulated signal used BPSK modulation, a symbol period of $T = 15$ μs, and a non-return to zero (NRZ) data format, also assumed known. The asterisk on Fig. 11-6 represents the estimator output, while the line (corresponding to $x = y$) is the desired output. With an observation of 100 samples, the SNR estimator is seen to perform quite well for this example.

11.3.4 Modulation Classification

The various modulation classifiers discussed in Chapter 9 each require knowledge of the data rate, pulse shape, and symbol timing in order to form a matched-filter output, i.e., the single-sample per symbol statistic $\tilde{r}_n$. That statistic is obtained by summing up L received samples, where L is the ratio of the sample rate to the symbol rate:

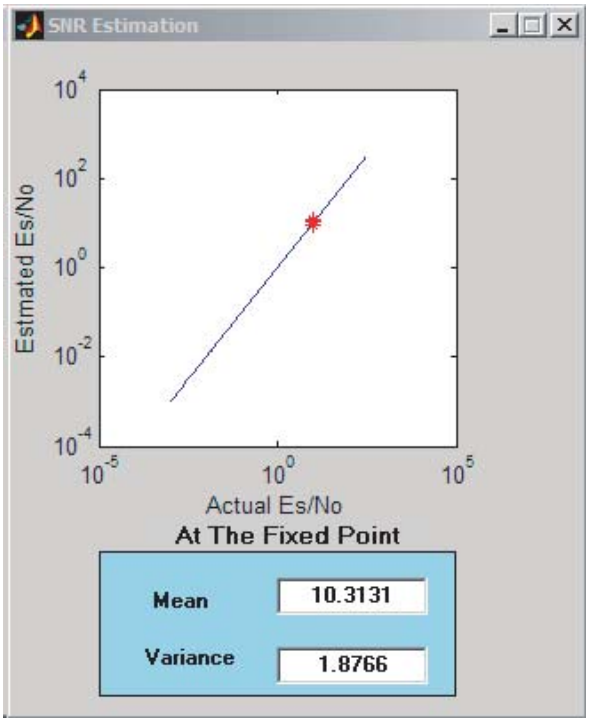


Fig. 11-6. Example output from an SNR estimator.

$$\tilde{r}_n = \sum_{k=(n-1)L+1}^{nL} \tilde{r}_d[k], \quad n = 1, 2, \cdots, N \tag{11-10}$$

with $\tilde{r}_d[k]$ given by Eq. (11-5).

The normalized quasi-log-likelihood ratio (nqLLR) modulation classifier does not need to know the SNR, which makes it a good candidate for modulation classification in the coarse estimation phase. However, the nqLLR classifier must still wait for the data rate and symbol-timing estimates to become available for Eq. (11-10) to be computed, at which point the SNR estimate is also available, since it comes from the same joint estimator. Therefore, there is no advantage in the coarse phase to using the nqLLR, even though in principle it requires less knowledge about the signal attributes.

Instead, the modulation classification in the coarse estimation phase is accomplished with the quasi-generalized-likelihood ratio test (qGLRT), discussed in Section 9.2.2. The qGLRT is based on the conditional-likelihood function $\mathrm{CLF}_M(\theta)$, given by Eq. (9-9) evaluated at a best-available estimate of θ_c. As shown in Chapter 9, this produces only a small loss compared to the ML classifier, which computes a full-blown average over θ_c:

$$\mathrm{LF}_M = \frac{M}{2\pi} \int_0^{2\pi/M} \mathrm{CLF}_M(\theta)\, d\theta \tag{11-11}$$

The estimate for θ_c is given by

$$\hat{\theta}_c^{(M)} = \frac{1}{M} \arg \sum_{n=0}^{N-1} \tilde{r}_n^M \tag{11-12}$$

For BPSK we have

$$\hat{\theta}_c^{(2)} = \frac{1}{2} \arg \sum_{n=0}^{N-1} \tilde{r}_n^2 \tag{11-13}$$

and for quadrature phase-shift keying (QPSK) we have

$$\hat{\theta}_c^{(4)} = \frac{1}{4} \arg \sum_{n=0}^{N-1} \tilde{r}_n^4 \tag{11-14}$$

and, thus, the qGLRT classifier metric is

$$\mathrm{LR} = \frac{\mathrm{CLF}_2\left(\hat{\theta}_c^{(2)}\right)}{\mathrm{CLF}_4\left(\hat{\theta}_c^{(4)}\right)} \tag{11-15}$$

If this is greater than unity, the modulation is declared to be BPSK; otherwise, the modulation is declared to be QPSK.

Figure 11-7 illustrates an example comparing the ML, generalized likelihood ratio test (GLRT), quasi-log-likelihood ratio (qLLR), and nqLLR modulation classifiers, when discriminating BPSK from QPSK. The performance is measured by probability of misclassification, which is the probability of deciding either BPSK or QPSK at the receiver when in fact the other modulation was transmitted. In the example, the symbol SNR is $E_s/N_0 = -4$ dB, and perfect residual frequency correction and symbol timing are assumed. As can be seen from the bar chart, even for such a low SNR, correct classification can still be accomplished about 90 percent of the time.

In the fine estimation phase, the phase tracking loop has locked onto the carrier phase θ_c. Thus, we switch to the coherent ML modulation classifier, which is the same as the qGLRT except that the phase estimate $\hat{\theta}_c$ coming from

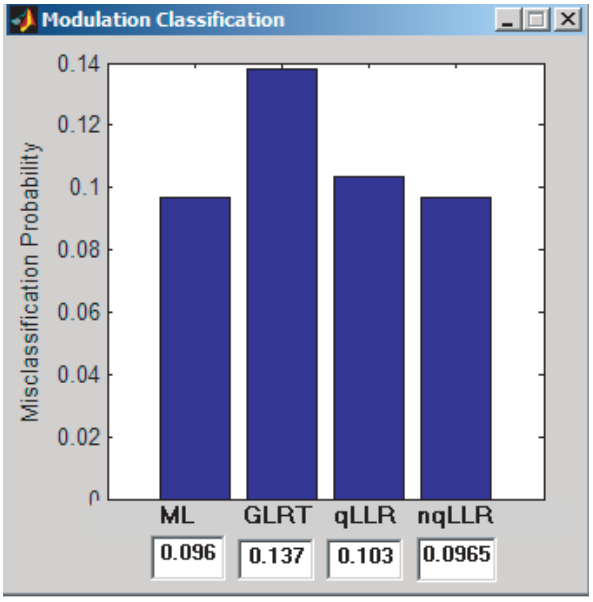


Fig. 11-7. Classification error probability of various classifiers for BPSK/QPSK, where $E_s/N_0 = -4$ dB, $\varepsilon = 0$, $\omega_r = 0$, using $N = 100$ observed symbols.

the carrier tracking loop is used in place of the GLRT estimate. In the case of BPSK/QPSK classification, Eq. (11-15) becomes

$$\mathrm{LR} = \frac{\mathrm{CLF}_2\left(\hat{\theta}_c\right)}{\mathrm{CLF}_4\left(\hat{\theta}_c\right)} \tag{11-16}$$

11.3.5 Carrier Synchronization

As shown in Fig. 11-2, carrier synchronization takes several forms, depending on the modulation index and coarse/fine operation.

If the modulation index estimator has determined that there is a residual carrier, then a phase-locked loop (PLL) may be used to lock onto the residual carrier signal. The residual carrier itself is the output of a low-pass filter of the received signal, which suppresses the data modulation (except for the portion of the spectrum at zero frequency). When a residual carrier is present, the PLL may be the best choice for carrier synchronization in the fine estimation mode as well, but this depends on the SNR and the value of the modulation index. In cases when the residual carrier is weak, a hybrid loop may outperform the PLL alone in the fine estimation phase.

If the modulation index estimator has determined that the carrier is suppressed, then the carrier synchronization in the coarse estimation phase relies on a carrier loop with passive arms. As discussed in Chapter 8, such a loop uses passive filters in each arm, so that the symbol timing and pulse shape need not be known. A universal loop can be constructed that will work for all M-PSK modulation orders up to some maximum $M_{\max}$. This is shown in Fig. 8-12 for $M \in \{2, 4, 8\}$, where the in-phase (I) and quadrature (Q) arm filters can be implemented with simple low-pass filters such as simple integrators. Since modulation classification is not yet available to the carrier loop during the coarse phase, the carrier loop begins in the coarse estimation phase configured for $M_{\max}$-PSK.

For suppressed carrier signals in the fine estimation phase, the switch for M in Fig. 8-12 can be set according to the modulation classifier output, and the passive arm filters can be replaced with matched filters that make use of the pulse shape and symbol-timing estimates, which results in improved performance as discussed in Chapter 8.

We now discuss the conversion of the passband continuous-time loops discussed in Chapter 8 to the discrete-time complex baseband loops suitable for a digital implementation. First we will discuss the continuous model for tracking a BPSK signal. For QPSK signals, the same approach applies.

Assume that the input to the loop is the BPSK suppressed carrier passband signal $r(t)$ in the form of

$$r(t) = s(t) = \sum_{n=1}^{N} a_n p\left[t - (n-1)\,T\right] \sin\left(\omega_c t + \theta_c\right) + n(t) \tag{11-17}$$

where a_n is random binary data corresponding to the nth transmitted BPSK symbol taking on values ± 1 with equal probability and for simplicity $p(t)$ is assumed to be a unit amplitude rectangular pulse. Let

$$\begin{aligned} w_s\left(t\right) &\triangleq \sin\left(\omega_c t + \hat{\theta}_c\right) \\ w_c\left(t\right) &\triangleq \cos\left(\omega_c t + \hat{\theta}_c\right) \end{aligned} \tag{11-18}$$

where $\hat{\theta}_c$ is the estimate of θ_c. Therefore, the nth transmitted BPSK symbol in the interval $(n-1)\,T \le t \le nT$ is given by

$$U = \int_{(n-1)T}^{nT} w_s\left(t\right) r\left(t\right) dt$$

Neglecting noise terms and double frequency terms results in

$$U \propto \cos\left(\theta_c - \hat{\theta}_c\right) \tag{11-19}$$

Similarly,

$$V = \int_{(n-1)T}^{nT} w_c\left(t\right) r\left(t\right) dt \propto \sin\left(\theta_c - \hat{\theta}_c\right)$$

Let $\phi \triangleq \theta_c - \hat{\theta}_c$. Then,

$$U \propto \cos\left(\phi\right) \tag{11-20}$$

$$V \propto \sin\left(\phi\right) \tag{11-21}$$

Now, consider $\tilde{r}_n\left[k\right]$, which is the input of the equivalent complex baseband, discrete-time loop. Assuming perfect carrier frequency estimation ($\omega_r = 0$),

$$\tilde{r}_n\left[k\right] = \frac{1}{L} e^{j\left[\theta_n + \theta_c\right]} + \tilde{n}_n\left[k\right] \tag{11-22}$$

The estimated signal $\hat{\tilde{r}}$ is the output of the discrete-time voltage-controlled oscillator (VCO) and has the form

$$\hat{\tilde{r}} = e^{-j\hat{\theta}_c} \tag{11-23}$$

where $\hat{\theta}_c$ is the estimate of θ_c. From Eqs. (11-22) and (11-23), the multiplication signal $\tilde{w}_n[k]$ has the form

$$\tilde{w}_n[k] = \tilde{r}_n[k] \times \hat{\tilde{r}} \tag{11-24}$$

$$= \left(\frac{1}{L} e^{j[\theta_n + \theta_c]} + \tilde{n}_n[k]\right) e^{-j\hat{\theta}_c} \tag{11-25}$$

$$= \frac{1}{L} e^{j\theta_n} e^{j\left[\theta_c - \hat{\theta}_c\right]} + \tilde{n}_n[k]\, e^{-j\hat{\theta}_c} \tag{11-26}$$

Since $\tilde{n}[k]$ and $\tilde{n}[k]\, e^{-j\hat{\theta}_c}$ have the same statistical distributions, we may write

$$\tilde{w}_n[k] = \frac{1}{L} e^{j(\theta_n + \phi)} + \tilde{n}_n[k] \tag{11-27}$$

The equivalent of integration in the continuous domain is summation in our domain. Therefore, the integrators are replaced with summations. Also note that, since we are dealing with complex signals, in order to extract the trigonometric functions (sine and cosine) of the signal angle, the loop requires extracting the imaginary and real parts, respectively, of the signal $\tilde{w}(k)$. Applying these modifications results in the loops that are illustrated in Figs. 11-8(a) and 11-8(b), where

$$S : \tilde{y}_n = \sum_{k=(n-1)L+1}^{nL} \tilde{x}_n[k]$$

Note that the loop filter is identical to the continuous case that is discussed in Chapter 8. Hence, the transfer function of the loop filter has the form

$$F(s) = \frac{1 + \tau_2 s}{\tau_1 s}, \quad \tau_1 \gg \tau_2 \Rightarrow (\tau_1 = 1, \tau_2 = 0.01) \tag{11-28}$$

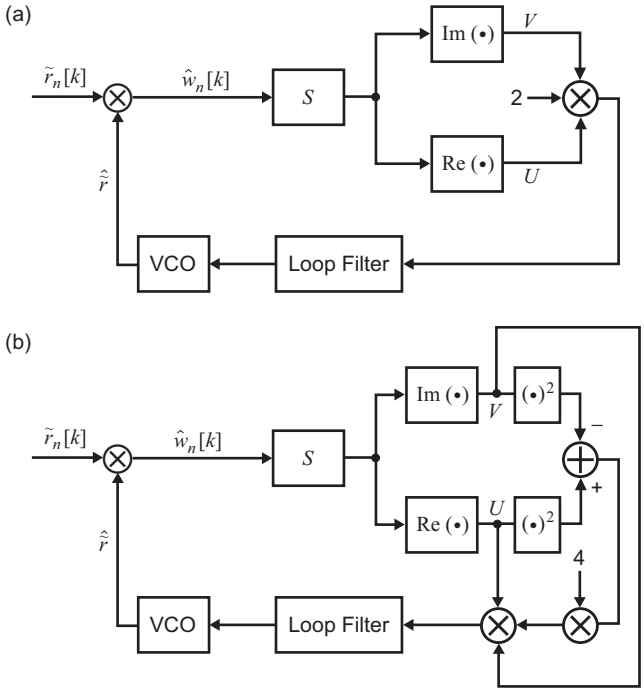


Fig. 11-8. Sampled implementation of a Costas-type loop capable of tracking: (a) BPSK and (b) QPSK.

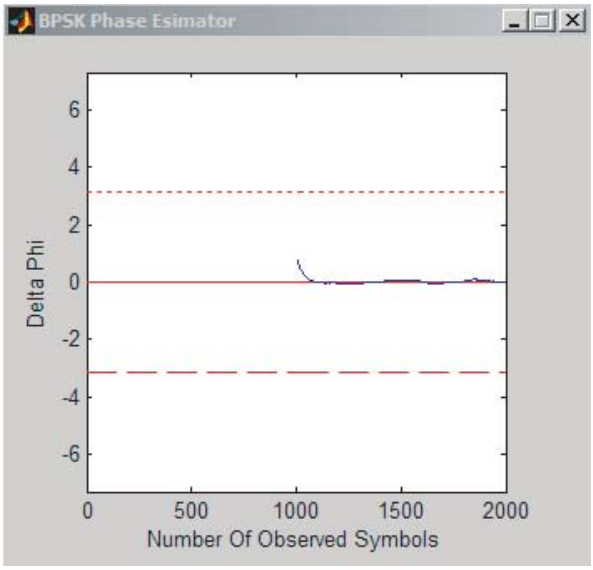
Figures 11-9(a) and 11-9(b) illustrate the sample acquisition behavior of each loop for a set of parameters. In each case, the phase error ϕ is shown as a function of time measured in terms of the number of observed samples at the receiver. As it is represented in the plots, the carrier phase tracking loop starts operating after the frequency correction is performed on 1000 symbols.

As can be observed, the loops do not always lock at zero phase error ($\phi = 0$). The reason is that, in BPSK and QPSK, the lock points of the loops are where $\sin(2\phi_{\text{BPSK}}) = 0$ and $\sin(4\phi_{\text{QPSK}}) = 0$, respectively, which result in an ambiguity of π for BPSK and $\pi/2$ for QPSK. Figures 11-9(a) and 11-9(b) show constant lines to indicate the other potential phases at which the carrier loop could lock.

11.3.6 Symbol Synchronization

In the coarse estimation phase, either the noncoherent cross-spectrum symbol synchronizer (CSSS) or the noncoherent DTTL may be used to acquire the symbol timing. The choice of which to use depends on the SNR, as discussed in Chapter 10, with the CSSS being preferred at low SNR. For either synchronizer, the symbol period and pulse shape must be known.

(a)



(b)

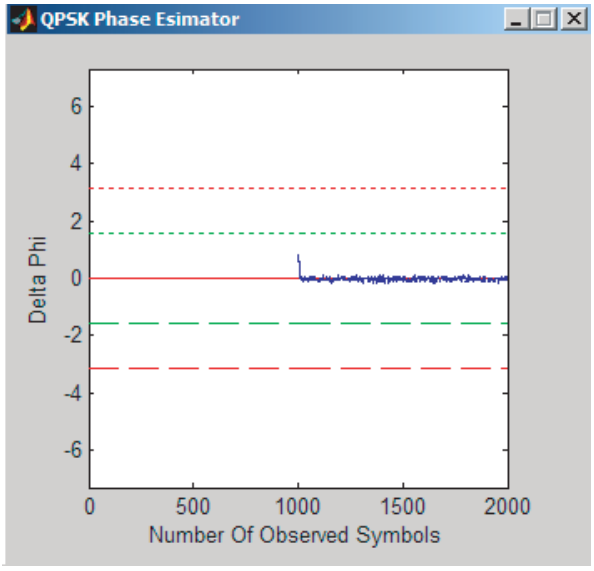


Fig. 11-9. Carrier synchronization loop performance: (a) BPSK and (b) QPSK.

In the fine estimation phase, a conventional coherent DTTL may be used. The remainder of this section discusses the method by which discrete-time matched-filter output samples are used by a DTTL. As discussed in the section on modulation classification, the complex observables corresponding to the matched-filter output at the time instants nL, where $n = 1, 2, \cdots, N$, are given by

$$\tilde{r}_n = \sum_{k=(n-1)L+1}^{nL} \tilde{r}_n\,[k], \quad n = 1, 2, \cdots, N \tag{11-29}$$

Thus,

$$\tilde{r}_n = \sum_{k=(n-1)L+1}^{nL} \left[\frac{1}{L} e^{j[\omega_r k T_s + \theta_n + \theta_c]} + \tilde{n}'_n\,[k]\right] \tag{11-30}$$

Therefore, the observation vector $\tilde{\mathbf{r}} = (\tilde{r}_1, \tilde{r}_2, \cdots, \tilde{r}_N)$ for a sequence of N symbols can be modeled as

$$\tilde{r}_n = e^{j[\omega_r k T_s + \theta_n + \theta_c]} + \tilde{n}'_n, \quad n = 1, 2, \cdots, N \tag{11-31}$$

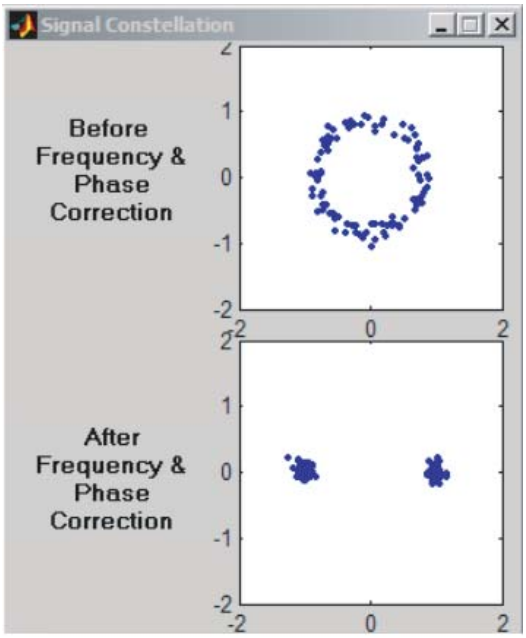
where $\tilde{n}'_n$ is a complex Gaussian random variable with mean zero and variance σ'^2 per dimension.

Figures 11-10(a) and 11-10(b) provide a visualization of the derotation of the signal constellation that takes place as a result of frequency and phase correction. The upper plot in each figure corresponds to the observation vector $\tilde{\mathbf{r}} = (\tilde{r}_1, \tilde{r}_2, \cdots, \tilde{r}_N)$, where $N = 100$ and a non-zero frequency offset (ω_r) was introduced to the system, and the lower plot represents the frequency- and phase-corrected version of $\tilde{\mathbf{r}}$.

Reference

[1] J. G. Proakis, *Digital Communications*, third ed., New York: McGraw Hill, Inc., 1995.

(a)



(b)

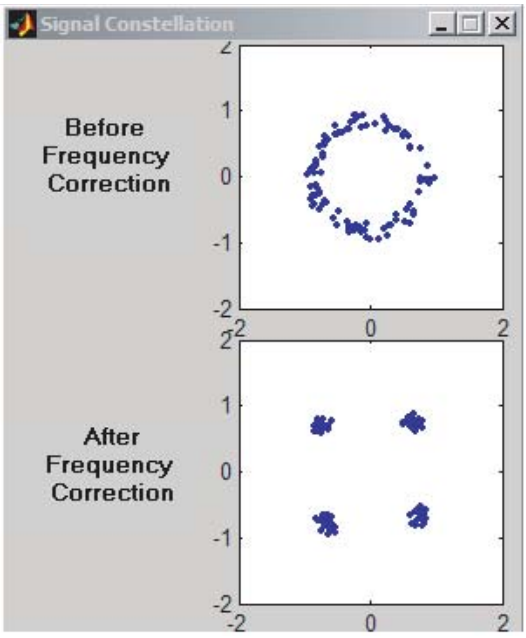


Fig. 11-10. Signal constellations for a signal with different modulations and the same parameters before and after frequency correction, with E_s/N_0 = 20 dB: (a) BPSK and (b) QPSK.